4.0 FILE QUERY PROCEDURES

This section discusses query capabilities both within and outside of JEPES.

4.1 System Query Capabilities

By selecting the Database Maintenance function, the user can issue queries to look at the database. There are some values and ranges the user must use to facilitate the query. Paragraph 5.3.2, Database Maintenance, describes the Database Maintenance retrieval operation and Appendix H, JEPES Codes, lists the various JEPES codes that enable a user to query a JEPES table.

4.2 Database

JEPES uses the ORACLE RDBMS to store data. Table 4.2-1, JEPES Tables, lists the 61 JEPES tables and views. Temporary tables or views are marked with a * prefix. See Appendix D, JEPES Data Element Dictionary, for a detailed description of the tables.

4.3 Query Preparation

This subsection discusses how the user may take advantage of the JEPES predefined SQL queries. **Note:** Advanced understanding of the SQL programming language and the JEPES RDBMS is required to use SQL independently.

JEPES execution is built on a series of *.s* files that invoke *.sql* files or *.ctl* files. The *.ctl* files are mainly used to load data from a flat file into existing ORACLE tables. The *.sql* files are called from the menu and used to issue queries to retrieve or update table information. It is not advisable to run these queries individually when they are not in JEPES since they may use a sequence of *.s*, *.ctl*, and *.sql* files, or they may coordinate with the execution of an Ada program. Running them independently may result in lost data (update/delete), mismatched data, or inaccurate retrievals. The following provides information on the use of predefined JEPES SQL queries.

To run a predefined *.SQL* query file, a user should do the following:

Go to Reports/Queries -> Ad Hoc option, and type:

@[directory path to desired SQL file]**/**[.sql file name]

example: **@JEPES_HOME/sql/nuc.sql**

Table 4.2-1. JEPES Tables (1 of 2)

MENU NAME	ORACLE TABLE NAME	TYPE
	Aggregated_Asset	table
ASSET	Asset	table
	Attrition_Factor	table
	*Avail	table
BACKUP SUPPLY	Backup_Supply	table
BASE COMPLEX	Base_Complex	table
BASE FACILITY CONSTRUCTION POLICY	Base_Fac_Construction_Policy	table
BASE LOCATION	Base_Location	table
	*Base_Sum	table
	Cargo_Aggregation_Period	table
	Climatic_Factor	table
	Component	table
COMBINED ASSET	Combined_Asset	table
COMPONENT EXCEPTION	Component_Exception	table
	Construction_Capability	table
DEPLOYED UNIT (TROOP)	Deployed_Eng_Sensitive_Unit	table
DESTINATION LOCATION	Destination_Location	table
ENGINEERING SUPPORT	Engineering_Support	table
ENGINEERING UNIT CAPABILITY	Engineering_Unit_Capability	table
EQUIPMENT PLANNING FACTOR	Equipment_Planning_Factor	table
EQUIPMENT TYPE	JEPES_Equipment_Type	table
FACILITY CATEGORY	Facility_Category	table
FACILITY CATEGORY SUBSTITUTE	Facility_Category_Substitute	table
FACILITY COMPONENT	Facility_Component	table
FACILITY REQUIREMENT	Facility_Requirement	table
GENERAL PLANNING FACTOR	General_Planning_Factor	table
	*Imp_Exp_List	table
	Keys	table
	LOGSAFE_Interface	table
	LSA_Export	table
	LSA_Interface	table
	LSA_Requirement	table
NON UNIT CARGO	Non_Unit_Cargo	table
OPERATION	Operation	table
ORIGINATING LOCATION	Originating_Location	table
	Phase_In_Efficiency	table
PLANNER INPUT REQUIREMENTS	Planner_Input_Requirement	table
PLAN FACILITY CONSTRUCTION POLICY	Plan_Fac_Construction_Policy	table
	*Plan_Sum	table
POD LOCATION	POD_Location	table
POE LOCATION	POE_Location	table
	*Preproj	table
	*Pre_Project	table
	*Pre_Unscheduled_Project	table

Table 4.2-1. JEPES Tables (2 of 2)

MENU NAME	ORACLE TABLE NAME	TYPE
PROJECT	Project	table
	Req_Analysis_Tracking	table
	Scheduled_Project	table
	Skill_Sub	table
	*S_P_Tab	table
	Time_Period	table
UNIT EQUIPMENT	Unit_Equipment	table
UNIT TYPE	JEPES_Unit_Type	table
	Unscheduled_Project	table
	Usr_Query	table
	Usr_Query1	table
WAR DAMAGE FACTOR	War_Damage_Factor	table
	*Adadata	view
	Geoloc_Tab	view
	*TP_POP	view
	*TP_Rqmts	view
	*V_2I	view

The following predefined queries are available within the JEPES to process reports and graphs:

a. Requirements Generation

1. The *req_dmp.sql* query: Creates the report from the Project table for the entire planning area.

2. The *req_base.sql* query: Creates the report from the Project table for a specific base complex. The user will be prompted to enter the BCN.

b. Requirements Analysis

1. The *asstst1.sql* query: Creates the report for all Asset Satisfied Requirements.

2. The *asstust1.sql* query: Creates the report for all Asset Unsatisfied Requirements.

3. The *asstst2.sql* query: Creates the report for Asset Satisfied Requirements for specific time/region constraints. Region/time constraints were defined by the Region/Time Constraints screen in the Requirements Analysis section. Refer to the TUG, see Paragraph 1.2, Project References.

4. The *asstust2.sql* query: Creates the report for Asset Unsatisfied Requirements for specific time/region constraints. Region/time constraints were defined by the Region/Time Constraints screen in the Requirements Analysis section. Refer to the TUG, see Paragraph 1.2, Project References.

5. The *cnstr1.sql* query: Creates the report for all Construction Satisfied Requirements.

6. The *cnstr2.sql* query: Creates the report for all Construction Requirements for the specific time/region constraints. Region/time constraints were defined by the Region/Time Constraints screen in the Requirements Analysis section. Refer to the TUG, see Paragraph 1.2, Project References.

7. The *cnstr3.sql* query: Creates the report for all Construction Requirements to be completed within the analysis period.

8. The *cnstr4.sql* query: Creates the report for all Asset Unsatisfied Requirements for specific region/time constraints. Region/time constraints were defined by the Region/Time Constraints screen in the Requirements Analysis section. Refer to the TUG, see Paragraph 1.2, Project References.

c. Non-Unit Cargo

1. The *nuc.sql* query: Creates Class IV requirements needed to support civil engineering requirements and store into the Non_Unit_Cargo table.

2. The *logsafe.sql* query: Extracts data from the Non_Unit_Cargo to add to the LOGSAFE_Interface table.

d. Logistic Sustainability Analysis

1. The *req_mrg.sql* query: Merges asset satisfied and unsatisfied requirements and adds them to the LSA_Requirement table.

2. The *v-2i-vw.sql* query: Gathers data used for producing the LSA graphs.